

筑波大学 情報学群 情報メディア創成学類

卒業研究論文

カテゴリデータの時間変化を
視覚的に表現する手法の開発

伊藤 聡

指導教員 三末 和男 志築 文太郎 田中 二郎

2012年2月

概要

カテゴリデータを分析することによって有益な知見を得ようとする試みは、様々な場面で行われるが、カテゴリデータが時間とともに変化するような場合、どのように変化するかを把握する必要がある。カテゴリデータの分析は、異なる属性同士の組み合わせに着目するために、表に基づいて行われることが多い。しかし、表に基づく分析ではデータの時間変化を把握することが難しい。

本研究では、時間変化するカテゴリデータにおいて、異なる属性同士の組み合わせを表現する手法を開発した。本手法では、データの時間変化を二次元平面上の図形として表現している。本手法を用いることにより、時系列によるデータの傾向や変化を、異なる属性の組み合わせに着目して、直感的に把握することができるようになった。

開発した手法を基に、カテゴリデータの時間変化を視覚的に分析するためのツールを開発した。実データを用いたユースケースを通して、本ツールを利用した分析が、カテゴリデータの時間変化から知見を得るために有用であることを示した。

目次

第 1 章	はじめに	1
1.1	カテゴリデータ	1
1.1.1	カテゴリデータとは	1
1.1.2	カテゴリデータの分析	1
1.2	時系列データ	2
1.2.1	時系列データとは	2
1.2.2	時系列データの分析	2
1.3	時間変化するカテゴリデータの分析における問題	3
1.4	本研究の目的	3
1.5	本研究の貢献	3
第 2 章	関連研究	5
2.1	カテゴリデータを対象とした手法	5
2.1.1	Parallel Coordinates を基にした手法	5
2.2	時系列データを対象とした手法	6
2.2.1	ThemeRiver を基にした手法	6
2.3	時系列情報を含むカテゴリデータを対象とした手法	6
第 3 章	要件定義	7
3.1	対象データ	7
3.2	データの定式化	8
3.3	分析における要件	8
第 4 章	表現手法	10
4.1	River の描画	11
4.1.1	Stream の幅の決め方	11
4.1.2	上端と下端の位置の決め方	11
4.1.3	River オーバーレイ	12
4.2	River の配色	14
4.2.1	River の各 Stream の配色	14
	色相パラメータの割り当て	14
	彩度パラメータの割り当て	14

第 5 章	分析ツール「TwinRiverViewer」の開発	16
5.1	ツールの概観	17
5.2	ツールの機能	17
5.2.1	データの視覚的表示	17
5.2.2	River の切り替え操作	17
5.2.3	Stream の選択操作	17
5.2.4	River オーバーレイ操作	19
5.2.5	ポップアップによるデータ件数の表示	19
5.3	実装言語とデータフォーマット	21
第 6 章	ユースケース	22
6.1	チケットデータ	22
6.1.1	分析に用いたデータ	22
6.2	分析の実例	23
第 7 章	今後の課題	27
7.1	三属性以上の組み合わせの表現	27
7.2	大規模データへの対応	27
第 8 章	まとめ	28
	謝辞	29
	参考文献	30

図目次

1.1	折れ線グラフの一例	3
3.1	時刻情報を持つクロス集計表の列	9
4.1	River の例	10
4.2	River の描画についての説明	11
4.3	River オーバーレイのイメージ	12
4.4	River オーバーレイについての説明	13
4.5	l 種類の属性への色相パラメータの割り当て	14
5.1	ツール全体の画面	16
5.2	一ヶ月ごとの縦線とラベルを非表示にした状態	18
5.3	表示する River を切り替えるコンボボックス	18
5.4	Stream を選択した状態: Stream の選択は右側の属性値リストからも行うことができる	19
5.5	River オーバーレイの状態: ある属性値を持つデータが、他の属性ではどのように分布しているかを見ることができる	20
5.6	ポップアップ: 通常時	21
5.7	ポップアップ: River オーバーレイ時	21
6.1	データを読み込んでから最初の表示	23
6.2	estimated_hours 属性の大半は、属性値なしであることが分かる	24
6.3	上: status 属性、下: project 属性	24
6.4	status 属性中の値「新規」の Stream を project 属性の River にオーバーレイ	25
6.5	status 属性中の値「新規」の Stream を assigned_to 属性の River にオーバーレイ	26

表 目 次

1.1	カテゴリデータの一例	1
1.2	クロス集計表の一例	2
1.3	時系列データの一例	2
3.1	時間とともに変化するカテゴリデータの一例	7
6.1	チケットの一例	22

第1章 はじめに

1.1 カテゴリデータ

実世界における情報を表すデータとして、カテゴリデータがその一つに挙げられる。また、実世界における情報は時間とともに変化するものが多く存在する。本研究の対象は、時間とともに変化するようなカテゴリデータである。

1.1.1 カテゴリデータとは

カテゴリデータとは、名義尺度の値を持つ変数によって構成されるデータである。表 1.1 にカテゴリデータの一例を示す。カテゴリデータが持つ変数は「属性」と呼ばれる。例えば、性別という属性は「男」と「女」という二種類の値で構成される。多くの場合、カテゴリデータは複数の属性を持つ多次元データである。

表 1.1: カテゴリデータの一例

性別	年齢	地域	支持政党
男	20代	茨城県	A党
女	30代	千葉県	B党
男	30代	埼玉県	A党
女	40代	東京都	C党
⋮	⋮	⋮	⋮

1.1.2 カテゴリデータの分析

個々のカテゴリデータを集計し、その結果を分析することで知見を得る試みは、数多く行われている。表 1.1 のデータを用いて行われる一例として、年齢別支持政党調査がある。

一般に、カテゴリデータの分析は特定の属性に着目して行われる。年齢別支持政党調査の場合は、「支持政党」と「年齢」という二つの属性に着目する。その集計結果は、表 1.2 のよ

うに表される。このような表は、行と列が交差する部分にそれぞれの集計結果が当てはまることから、クロス集計表と呼ばれる。クロス集計表は、行と列にそれぞれ属性を当てはめることによって、異なる二つの属性に着目した情報を表すことができる。

表 1.2: クロス集計表の一例

	20代	30代	40代
A党	750	700	500
B党	300	400	600
C党	150	100	100

1.2 時系列データ

1.2.1 時系列データとは

時系列データとは、データの持つ値が時間の経過とともに変化していくデータのことである。実世界で頻繁に扱われるデータであり、都市の人口、地点の気温、商品の販売量などは時系列データである。表 1.3 に、時系列データの一例を示す。

表 1.3: 時系列データの一例

年	2008年	2009年	2010年	2011年
販売数	10万	20万	30万	30万

1.2.2 時系列データの分析

時系列データの分析においては、データの特性が時間によってどのように変化したかを把握することが必要となる [1]。時間によるデータの変化を表現する手法の一つとして、折れ線グラフが挙げられる。折れ線グラフは、時間と量を軸とする平面上にプロットした点を線で繋ぎ、時間の経過に伴うデータの変化を視覚的に表している。図 1.1 に、折れ線グラフの一例を示す。

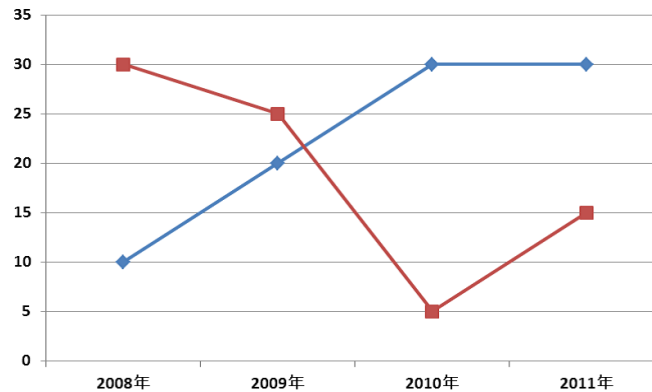


図 1.1: 折れ線グラフの一例

1.3 時間変化するカテゴリデータの分析における問題

実世界におけるカテゴリデータは、時間の経過とともに値が変化するという、時系列データの特徴を合わせ持つものも多く存在する。例として挙げた表 1.1 のデータも、通常は年月の経過とともに変化していく。このようなデータを分析するには、複数のクロス集計表を時系列順に並べて、それぞれを見比べる必要がある。しかし、膨大な数の表からデータの傾向や変化を読み取ることは困難である。

時系列データを視覚的に表すアプローチとしては、折れ線グラフや積み上げ面グラフなどが存在する。しかし、これらのグラフは特定の一属性に着目しなければならず、カテゴリデータの分析に必要な、異なる複数の属性への着目ができないという問題がある。

1.4 本研究の目的

本研究では、時間とともに変化する多次元のカテゴリデータを分析し、知見を得ることを支援する。これを実現するために、時系列データを表現する手法の一つである ThemeRiver[2] を基にした表現手法を開発する。この手法は、一属性の時間変化に加えて、異なる二属性に着目して時間変化を表すことが可能である。また、本手法を用いて、時間とともに変化するカテゴリデータの分析を支援するためのツールを開発する。

1.5 本研究の貢献

本研究の貢献は、以下の二点であると考えている。一つ目は、カテゴリデータの時間変化を異なる属性に着目して、視覚的に表現できるような手法を提案したことである。本手法により、カテゴリデータの時間変化の傾向や過程を、直感的に把握することが可能となる。

二つ目は、本手法を用いたカテゴリデータの視覚的な分析ツールを開発したことである。本ツールにより、時間とともに変化するカテゴリデータの分析を、異なる属性の組み合わせに着目して、視覚的に行うことが可能になると考えられる。

第2章 関連研究

カテゴリデータや時系列データを対象とした様々な可視化研究が行われている。まず、カテゴリデータ、時系列データそれぞれを対象とした研究を紹介し、次に時系列情報を含むカテゴリデータを対象とした研究を紹介する。

2.1 カテゴリデータを対象とした手法

カテゴリデータを対象とした、情報を視覚的に表現する研究は多く行われている。

Mosaic plots[3] は、矩形を用いてカテゴリデータを表現する手法である。度数の割合を基に矩形を分割し、それぞれの矩形を平面上に並べることで、視覚的にカテゴリデータを表すことができる。

FacetMap[4] は、インタラクティブな操作によってカテゴリデータを表現する手法の一つである。各属性を「ファセット」と呼称し、属性の集合を円や長方形などの図形で表現している。ファセット毎の絞り込みを、インタラクティブな操作で行える点が特徴的である。

つぶつぶビュー [5] は、「つぶつぶ表現」によってカテゴリデータのインタラクティブな分析を行うツールである。カテゴリデータにおける各エンティティを円として表すことによって、各属性の集合を視覚的に捉えることが可能となっている。

2.1.1 Parallel Coordinates を基にした手法

Parallel Coordinates[6] は、カテゴリデータのように複数の属性を持つデータを視覚化する手法の一つとして、広く知られているものである。各属性を表す座標軸を用意し、それぞれの座標軸上の点を線で結んでいくことで、データセット全体を表現することができる。Parallel Coordinates を基に、様々な表現手法が研究されている。

Parallel Sets[7] は、着目した属性によって線の色分けや太さを変化させることにより、データセットにおける属性同士の関係を視覚的に表現している。また、インタラクティブな手法によって、各属性に対して様々な操作を行うことができる。

Parallel Tag Clouds[8] は、Parallel Coordinates と Tag Clouds を組み合わせることによって、属性間におけるデータの変化が視覚的に把握できるようになっている。Tag Clouds を一列に表示した座標軸を並べ、異なる属性に存在する同じ値を線で結ぶことにより、データの変化を表現している。

2.2 時系列データを対象とした手法

時系列データを対象とした、情報を視覚的に表現する研究も多く行われてきている。

CloudLines[9] は、動的なイベントデータをベースとした時系列データの可視化手法である。時系列上の個々のイベントを円で表し、円の存在する密度を大きさや色の濃度に反映することで、時系列上の重要なイベントを表現している。

また、時間軸を螺旋状に描画し、データの値の変化を色や点で表現する手法 [10] も提案されている。この手法では時間軸を螺旋状にすることで、周期的な変化のパターンを表現することができる。

2.2.1 ThemeRiver を基にした手法

ThemeRiver は、時系列データの時間変化を表す手法の一つである。時間を表す座標軸を用意し、それぞれの座標軸上にプロットした点を曲線で結ぶ。各座標軸における曲線と曲線の間の長さが量に対応し、これによってデータの時間変化を表現している。この手法を基にした多くの表現が研究されている。

Visual Backchannel[11] は、大規模なユーザの集合における話題の変化を見る手法である。時系列上での話題の流れを表現する手法として、Topic Streams を提案しており、話題の時間変化を ThemeRiver のように把握することが可能である。

TextFlow[12] は、テキストドキュメントを対象とした、時系列上のトピックの発展を分析するツールである。この手法では、ThemeRiver にグラフィケイアウトを組み合わせることにより、増加や減少に加えて、枝分かれや合流も視覚的に表現することが可能となっている。

2.3 時系列情報を含むカテゴリデータを対象とした手法

時系列情報を含むカテゴリデータは、時間とともに変化するようなカテゴリデータであり、本研究が対象とするデータである。このデータを対象とする研究もいくつか行われている。

SellTrend[13] は、航空便のチケット予約および購入のデータを対象とした、カテゴリデータ分析ツールである。Treemap と時系列棒グラフを組み合わせたインタフェースによって、カテゴリデータの時間による変化を表現している。

ColorWave[14] は、プロジェクト管理ツールにおけるチケットデータの視覚的な分析を支援するツールである。数千件から数万件におよぶチケットデータのイベントや属性値を一画面で表現し、それらをインタラクティブに操作することが可能となっている。ColorWave が対象としているチケットデータは、時系列を含むカテゴリデータとして考えることができる。

第3章 要件定義

本研究が対象とするデータについて述べ、分析において必要となる要件を整理する。

3.1 対象データ

本研究で分析対象となるデータには、以下のような特徴があると考ええる。

1. 時刻の情報を持つ
2. 時刻以外に複数の属性を持つ
3. 時刻以外の属性には名義尺度の値が当てられる
4. 異なる時刻では属性の値が異なる場合がある

第 1.1 節で述べたように、カテゴリデータは名義尺度の値を持つ属性によって構成されるデータである。本研究で対象とするデータは複数の属性を持ち、それらの値が時間とともに変化していくような項目からなる、多次元のカテゴリデータである。表 3.1 に、対象とするデータの一例を示す。表全体が一つの項目を表しており、この例では個人に相当する。「支持政党」「性別」「年代」が属性であり、「A 党」「B 党」「C 党」は属性「支持政党」の値である。また、データに時刻の情報が含まれ、時刻の変化とともに属性「支持政党」の値が変化している。

表 3.1: 時間とともに変化するカテゴリデータの一例

時刻	支持政党	性別	年代	...
2008年11月	A党	男	20代	...
2009年5月	B党	男	20代	...
2009年11月	C党	男	30代	...
⋮	⋮	⋮	⋮	⋮

3.2 データの定式化

表 3.1 を例にとって、本研究が対象とするデータの定式化を行う。項目の全体集合を U で表す。 U の要素は、個々の項目である。

$$U = \{u_1, u_2, \dots, u_k\} \quad (3.1)$$

項目は複数の属性を持ち、それぞれの属性は取りうる値を持っている。また、属性のとり値は時間とともに変化する。このことを考慮し、属性 s_i を以下のような関数で表す。

$$s_i : U \times T \rightarrow V_i \quad (i = 1, 2, \dots, l) \quad (3.2)$$

$$V_i = \{v_{i,1}, v_{i,2}, \dots, v_{i,m_i}\} \quad (i = 1, 2, \dots, l) \quad (3.3)$$

$$T = \{t_0, t_1, \dots, t_n\} \quad (3.4)$$

ここで、 V_i は属性 s_i が取りうる値の集合、 T は時刻の集合とする。属性 s_i は項目の集合 U から属性値の集合 V_i への関数として表現できるが、時間とともに属性のとり値は変化するため、項目と時刻の直積集合から属性値への関数で表すこととした。つまり、 $s_i(u, t)$ は、項目 $u \in U$ の時刻 t における属性値を表す。表 3.1 における「支持政党」を s_1 とすると、その値である「A 党」「B 党」「C 党」が V_1 の各要素に相当する。このとき、 $s_1(u, t)$ は、 u の時刻 t における支持政党を表すことになる。

3.3 分析における要件

第 1.1 節で述べたように、カテゴリデータの分析を行う際には、異なる属性同士の組み合わせに着目して情報を得る事が多い。また、時系列データの分析には、時間によるデータの変化の傾向や過程を把握することが求められる。

時間変化するカテゴリデータの分析支援は、この両方を要求されることになる。すなわち、データの持つ異なる属性同士の組み合わせに着目しつつ、データの傾向や分布が変化する様子を把握できるようにする必要がある。

異なる属性同士の組み合わせに着目可能で、時間によるデータの変化を表すような方法としては、図 3.1 のような表の並びが考えられる。これは、時刻情報を持つクロス集計表を時系列順に並べたものであり、「支持政党」と「年代」の二属性に着目している。

しかし、これらの表から情報を得るためには膨大な量のセルを読み取る必要があり、直感的にデータを把握することは難しい。したがって、時間による量の変化を視覚的に提示し、直感的にデータが把握できるようにする必要がある。さらに、データを異なる属性間で比較することも分析においてよく行われるため、比較のための表現や操作体系が必要となる。

2009年5月	20代	30代	40代	...
2008年11月	20代	30代	40代	...
2008年5月	20代	30代	40代	...
A党	750	700	500	...
B党	300	400	600	...
C党	150	100	100	...
⋮	⋮	⋮	⋮	⋮

図 3.1: 時刻情報を持つクロス集計表の列

以上のようなデータ分析の特徴を踏まえ、時間とともに変化するカテゴリデータの分析における要件は、以下のようなものであると考える。

1. 時間による量の変化を直感的に把握できること
2. 異なる属性同士の組み合わせに着目できること
3. 着目する属性を変えてデータの比較ができること

第4章 表現手法

第 3.3 節で述べた要件を基に、カテゴリデータの時間変化を視覚的に表す手法を開発した。この手法は、ThemeRiver を基にしたもので、異なる属性の組み合わせに着目しながら、データの量的な傾向を視覚的に表現することができる。

本手法においては、時系列上での量の変化が図 4.1 のような図形で表される。本論文では、このような図形を「River」と呼称する。River は横に時間、縦に項目の件数を対応させた二次元平面上に描画される。また、一つの River は一つの属性に対応しており、各属性の値に対応する領域に分割されている。本論文では、これらの領域を「Stream」と呼称する。図 4.1 では、4 つの Stream によって River が構成されている。River と Stream を用いることにより、時間による量の変化を直感的に把握することが可能となり、分析要件 1 を満たすことができる。

本章では、第 3.2 節で行った定式化を基に、開発した表現手法について説明する。

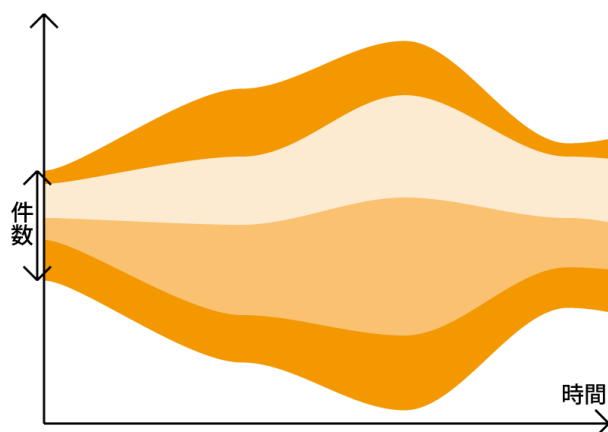


図 4.1: River の例

4.1 River の描画

4.1.1 Stream の幅の決め方

図 4.2 を例にとって、River における Stream の幅の決め方について説明する。図 4.2 は、4 種類の値を持つような属性を表現した River の例である。この River は、属性が持つそれぞれの値に対応した 4 つの Stream からなる。

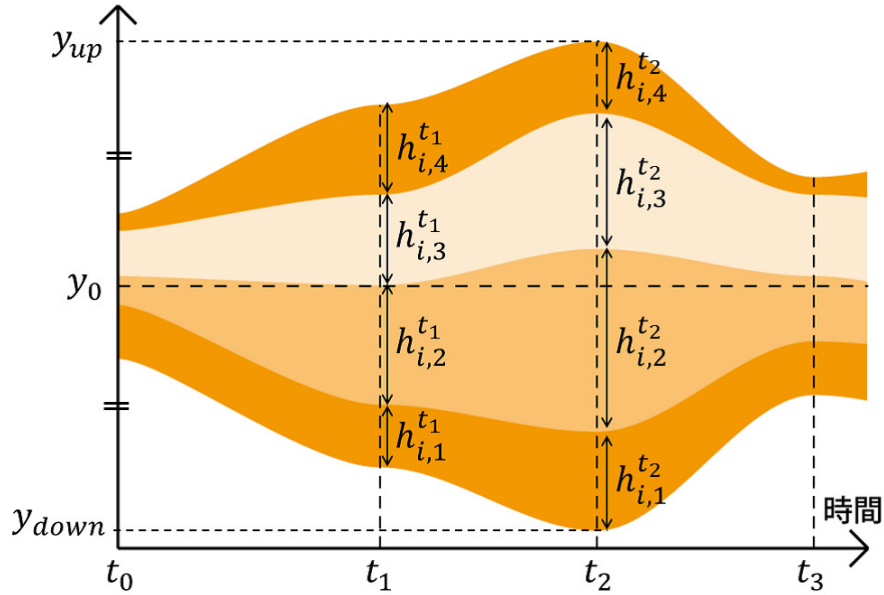


図 4.2: River の描画についての説明

t_0 は時間軸の始点である。 t_1 から t_3 はそれぞれ、特定の時刻を表している。属性が持つ値が 4 種類なので、River が 4 つの Stream に分割され、色分けされる。

属性 s_i に対応する River において、属性値 $v_{i,j}$ に対応する Stream の時刻 t における幅を、 $h_{i,j}^t$ で表すことにする。幅 $h_{i,j}^t$ は、式 4.1 で求められる。

$$h_{i,j}^t = c \cdot |\{u \in U \mid s_i(u, t) = v_{i,j}\}| \quad (4.1)$$

つまり、時刻 t において属性 s_i の値が $v_{i,j}$ であるような、項目の件数に比例した幅になる。 c は、River が描画領域からはみ出ないように設定される定数である。対応する属性値を持つデータがない場合は、Stream の幅は 0 となり、その時刻では見えなくなる。

4.1.2 上端と下端の位置の決め方

River の全体的な形は、中心軸 y_0 について線対称になる。そのために、以下の式 4.2 が全ての時刻において成り立つように、River の上端と下端の位置が決定される。 y_{up} は River の上

端、 y_{down} は River の下端を表す。

$$y_{up} - y_0 = y_0 - y_{down} \quad (4.2)$$

River の y_{up} と y_{down} は、式 4.3 および 4.4 によって求められる。

$$h_i^t = \sum_{j=1}^{m_i} h_{i,j}^t$$

$$y_{up} = y_0 + \frac{h_i^t}{2} \quad (4.3)$$

$$y_{down} = y_0 - \frac{h_i^t}{2} \quad (4.4)$$

m_i は River を構成する Stream の数であり、属性が持つ属性値の総数である。 h_i^t の値は、River の幅である。すなわち、 y_{up} は中心軸から正方向に River の幅の半分だけ移動した点であり、 y_{down} は同じ距離を負方向に移動した点である。 h のとる値は時刻によって変動するため、 y_{up} と y_{down} の値も時刻によって変動する。

4.1.3 River オーバーレイ

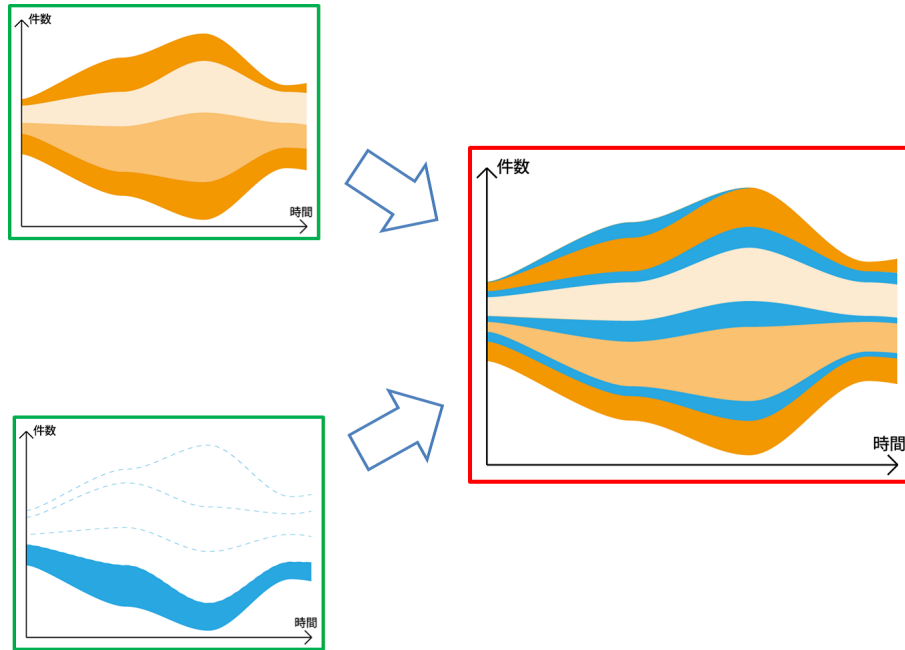


図 4.3: River オーバーレイのイメージ

River と属性は一对一の対応関係にあり、一つの River は一つの属性を表す。これに加えて、異なる属性同士の組み合わせを表現するため、River オーバーレイという手法を考案した。こ

の手法は、ある属性を表す River 中の Stream を指定し、他の属性を表す River の各 Stream に合わせて、分割して描画する（図 4.3）。これによって、異なる属性同士の組み合わせを表現することが可能となり、第 3.3 節の分析要件 2 を満たすことができる。

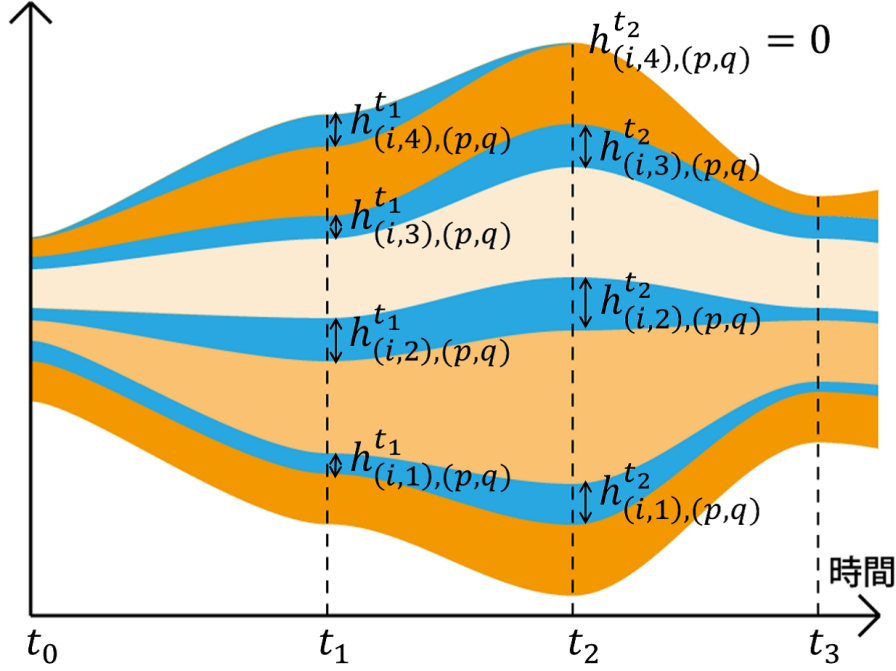


図 4.4: River オーバーレイについての説明

図 4.4 を例にとって、River オーバーレイの描画方法について説明する。図 4.4 は、図 4.2 の River 上に、別の River の Stream を一つオーバーレイした場合の例である。River オーバーレイの Stream は、オーバーレイ先の各 Stream の上端に合わせて描画される。 t_0 および t_1 から t_3 が表すものは、図 4.2 と同様の時刻である。

属性値 $v_{p,q}$ をオーバーレイ元に指定したとき、時刻 t において属性値 $v_{i,j}$ に対応する Stream にオーバーレイされる Stream の幅を、 $h_{(i,j),(p,q)}^t$ で表すことにする。幅 $h_{(i,j),(p,q)}^t$ は、式 4.5 で求められる。

$$h_{(i,j),(p,q)}^t = c \cdot |\{u \in U \mid s_i(u, t) = v_{i,j} \wedge s_p(u, t) = v_{p,q}\}| \quad (4.5)$$

オーバーレイする Stream の幅は、各 Stream に対応する属性値と指定した属性値を同時に持つような、項目の件数に比例する。属性値 $v_{p,q}$ を変えることに応じて、River オーバーレイの Stream も変化する。また、River オーバーレイの Stream において、同時刻での幅を全て合計すると、River オーバーレイ元に指定した属性値に対応する Stream の幅 $h_{p,q}^t$ と一致する。

4.2 River の配色

4.2.1 River の各 Stream の配色

River における Stream の描画に用いる色は、River の対応する属性と、各 Stream の River 内における並び順によって決定される。色相パラメータは属性によって決まり、彩度パラメータは並び順によって決まる。また、透明度パラメータは常に 1 であり、色の透過は行わない。これは色の透過を行った場合、背景色が River や Stream の色に影響を与えてしまうためである。

色相パラメータの割り当て

属性 s_i に対応する River における色相パラメータ $C_h(i)$ は、次の式 4.6 で決定される。

$$C_h(i) = \frac{i-1}{l}, \quad (i = 1, 2, \dots, l) \quad (4.6)$$

色相を属性の数で等分し、順番に各属性に一つずつ割り当てている（図 4.5）。これによって、二つ以上の属性で同じ色相の色が使われない。このことは River オーバーレイ時に特に有効であり、属性を色で区別することができる。図 4.5 は、 l 種類の属性が存在する場合における、色相パラメータの割り当て方を表す図である。

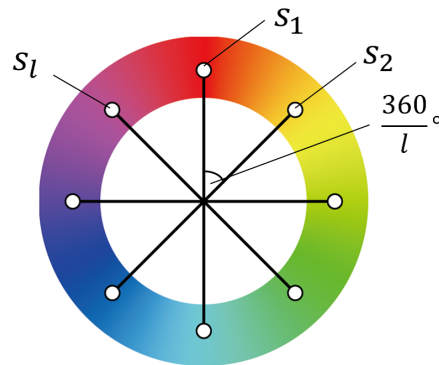


図 4.5: l 種類の属性への色相パラメータの割り当て

彩度パラメータの割り当て

River 領域における彩度パラメータは、Stream が River 内で何番目に並んでいるかによって決定される。順番は、幅が 0 の Stream も含めて下から順に数える。彩度は三段階に分けられ、一番目が 1、二番目が $\frac{2}{3}$ 、三番目が $\frac{1}{3}$ となる。四番目の Stream には一番目と同じ彩度が割り当てられ、以降同様の割り当て方が繰り返される。

色相のように Stream の数で彩度を等分し、各 Stream に順番に割り当てる方法も考えられる。しかし、その方法では隣接する彩度の差が小さいために、Stream 同士の区別が付かなくなることが多かった。彩度の分け方を三等分に定めることで、隣接する Stream を判別しやすくなることができた。

第5章 分析ツール「TwinRiverViewer」の開発

第4章での提案手法を用いた視覚的な分析支援ツールとして、「TwinRiverViewer」を開発した。図5.1は、本ツールの全体図である。第3.3節にて提示した分析要件を踏まえながら、視覚的な分析を支援する機能を紹介する。

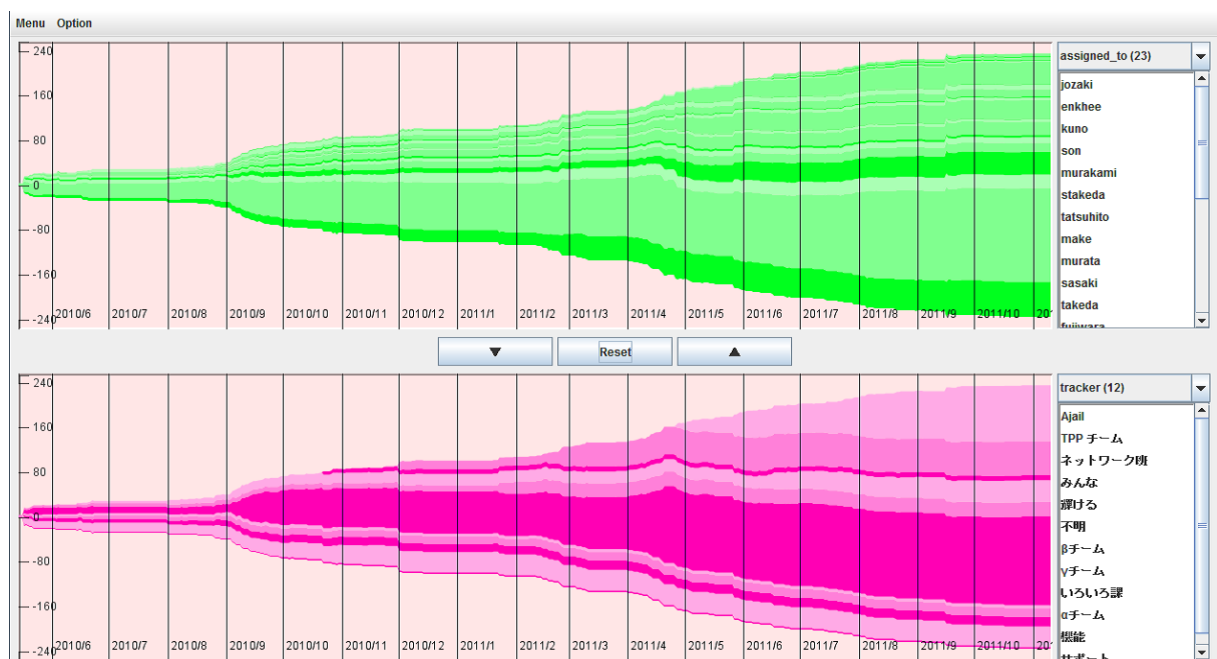


図 5.1: ツール全体の画面

5.1 ツールの概観

本ツールでは、カテゴリデータにおける各属性の傾向や時間による変化を、先述の表現手法を用いて提示する。左側には描画領域が配置されており、読み込んだデータが River として表示される。右側には、表示する River の属性を切り替えるコンボボックスや、現在表示されている属性に対応する属性値を表示するリストが配置されている。これらが上下に二つ並んだパネルに配置され、その間には River オーバーレイを行うためのボタンが配置されている。

上下にはそれぞれ異なる属性の River を表示できる。これにより、着目する属性を変えてデータ同士の比較を行うことが可能となり、第 3.3 節の分析要件 3 を満たすことができる。

5.2 ツールの機能

5.2.1 データの視覚的表示

描画領域には、読み込まれたカテゴリデータの一属性が先述の表現手法を用いて表示される。これによって、データの時間変化の傾向や過程を視覚的に読み取ることができる。

また、時系列上の時間位置を示すために、月の初めを表す縦線とラベルを描画している。この縦線とラベルは表示、非表示を切り替えることが可能である。図 5.2 は、縦線とラベルを非表示にした状態である。縦線を非表示にすると、River や Stream の全体的な形がはっきりと見えるようになるが、時間位置の把握はし辛くなる。

5.2.2 River の切り替え操作

描画領域の右側には、図 5.3 のようなコンボボックスが配置されている。ここから属性名を選択することで、どの属性の River を表示するかを切り替えることができる。上下の描画領域では、それぞれ異なる River を選ぶことができ、異なる属性についてのデータ傾向を対比しながら見ることができる。また、コンボボックスには属性の名前と共に、その属性が取りうる値の数も表示される。

5.2.3 Stream の選択操作

Stream の選択は、後述する River オーバーレイの指定に用いる他、特定の属性値に注目して見たい場合の強調表示として利用する。

描画領域上の各 Stream は、マウスオーバーされると色の輝度パラメータが変化し、ハイライトされる。この状態でクリックすることによって、River を選択することができる。選択状態にある Stream は常にハイライトされるとともに、輪郭部分が白色で描画される。図 5.4 は、assigned_to 属性の値「kamo」に対応する Stream を選択した状態である。

また、Stream は右側の属性値リストから選択することも可能である。リスト内の属性値を選択することによって、その属性値に対応する Stream を選択することができる。Stream と属

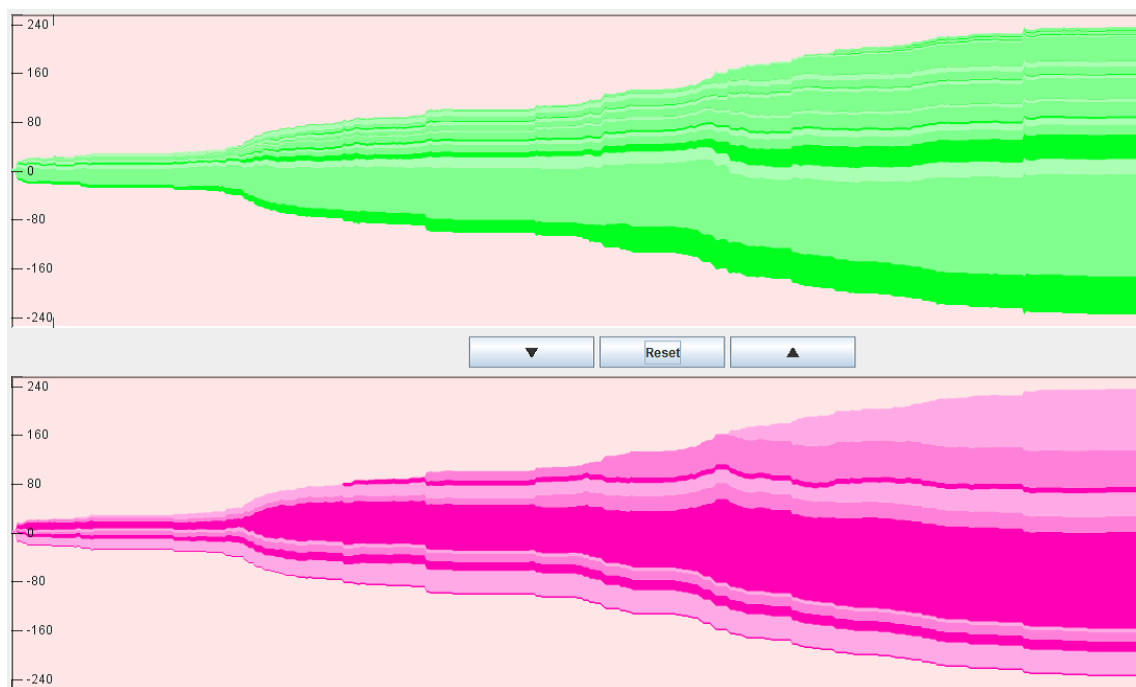


図 5.2: 一ヶ月ごとの縦線とラベルを非表示にした状態

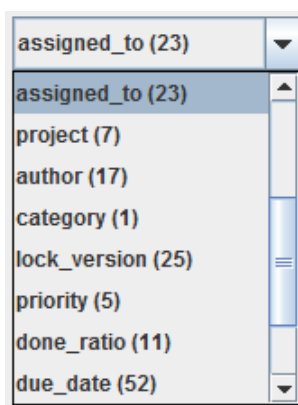


図 5.3: 表示する River を切り替えるコンボボックス

性値リストの選択状態は連動しており、描画領域側で選択が行われると、属性値リスト側の選択状態も変化する。

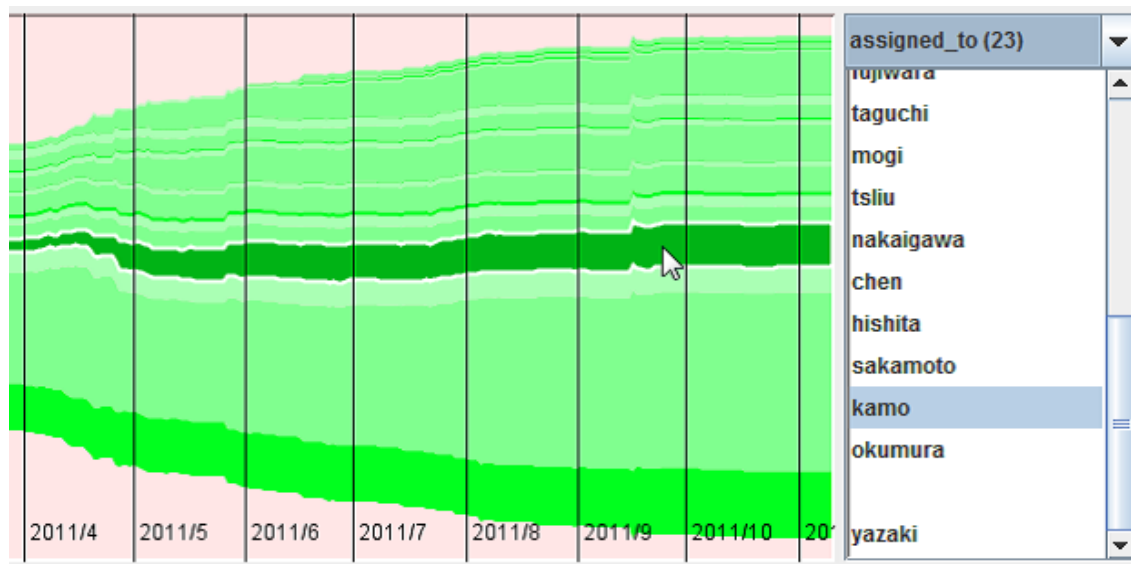


図 5.4: Stream を選択した状態: Stream の選択は右側の属性値リストからも行うことができる

5.2.4 River オーバーレイ操作

画面中央に配置された三つのボタンは、River オーバーレイ操作を行うボタンである。下向き三角の書かれたボタンは、上の River で選択された Stream を、下の River にオーバーレイする。逆に、上向き三角の書かれたボタンは、下の River で選択された Stream を、下の River にオーバーレイする。Reset と書かれたボタンは、表示されている River オーバーレイを消去する。Stream の River オーバーレイ状態は、別の Stream で新たに River オーバーレイを行うか、Reset ボタンで River オーバーレイを消去するまで継続される。

River オーバーレイの指定元となる属性値は、それぞれ選択されている Stream に対応する属性値である。対応する属性とその値は、River オーバーレイ中の描画領域の左上に表示される。これによって、特定の属性値が他の属性において、どのように分布しているかを把握することができる。図 5.5 は、River オーバーレイの一例である。

5.2.5 ポップアップによるデータ件数の表示

描画領域上の Stream をマウスオーバーすると、マウスポインタの位置にカーソルが表示される。ここで、カーソル位置に対応した時刻、およびその時刻における該当データの件数が、Stream に対応する属性値とともにポップアップ表示される。これによって、ある属性値のデー

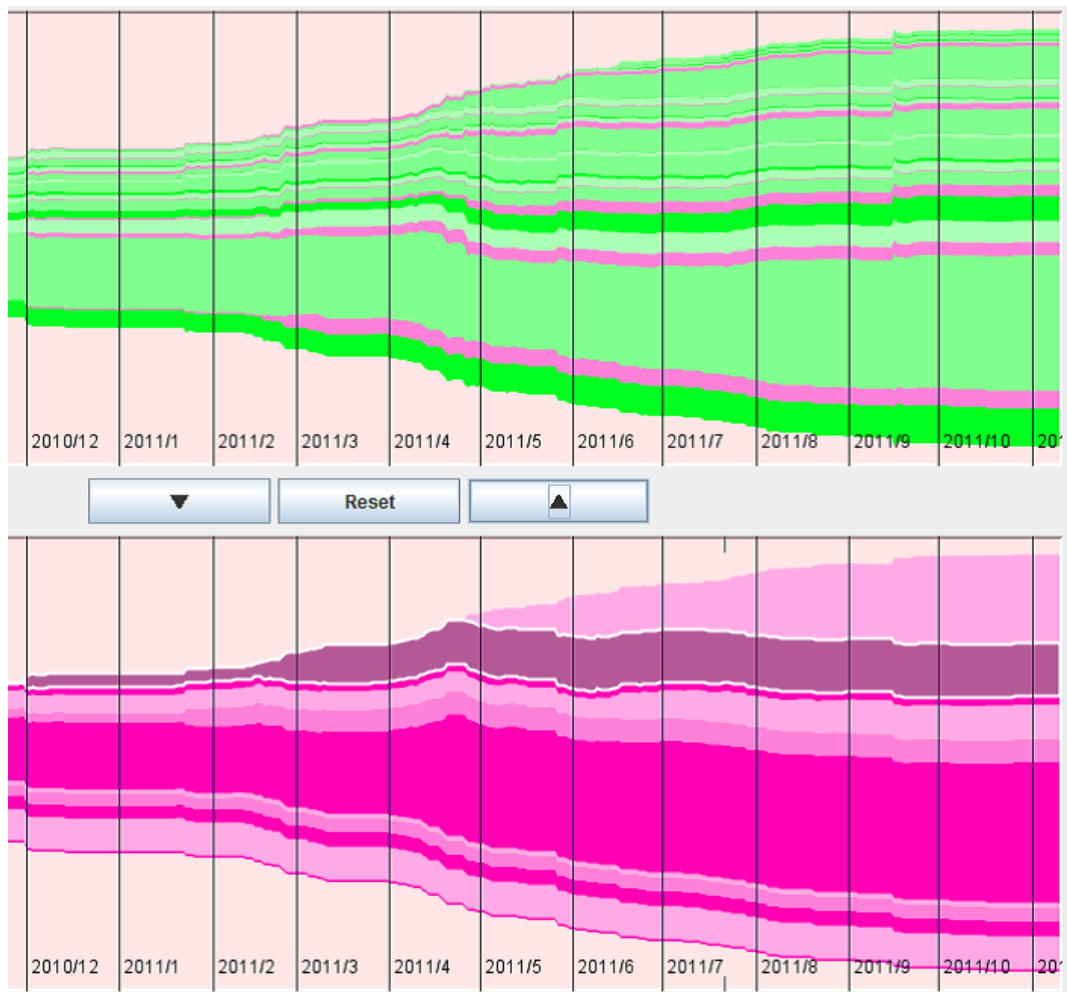


図 5.5: River オーバーレイの状態: ある属性値を持つデータが、他の属性ではどのように分布しているかを見ることができる

タがどの時間に何件存在するかという、具体的な情報を知ることができる。図 5.6 は、ポップアップ表示の一例である。この図では、2011 年 7 月 25 日の時点で属性値「Ajail」を持っているデータが 63 件存在することを示している。

River オーバーレイ状態では、オーバーレイしている Stream に対応する該当件数も共に表示される。図 5.7 はその一例である。図 5.6 での該当データのうち、さらに属性値「yazaki」を持っているデータが、63 件中 5 件存在することを示している。

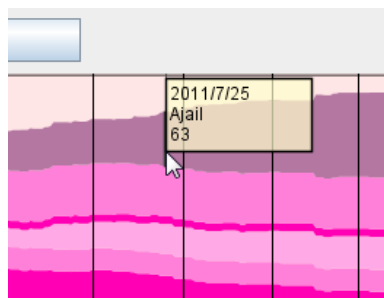


図 5.6: ポップアップ: 通常時

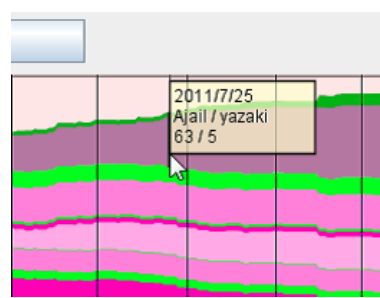


図 5.7: ポップアップ: River オーバーレイ時

5.3 実装言語とデータフォーマット

本ツールの実装言語は Java を使用しており、Java Runtime Environment¹上で動作する。ツールに読み込むデータのフォーマットは、JSON²形式を用いている。また、JSON 形式のファイルを Java で利用するためのライブラリ³も使用している。

¹<http://java.com/>

²<http://json.org/>

³<http://www.json.org/java/>

第6章 ユースケース

分析で利用したデータについて説明するとともに、本ツールを用いた分析の一例を示す。

6.1 チケットデータ

分析に使用したデータは、チケットデータと呼ばれるデータセットである。チケットデータは、活動の進捗情報をタイムスタンプとともに記録したもので、redmine¹等のプロジェクト管理ツールで用いられている。チケットはデータ中に複数の属性を持ち、タスクの進捗とともに属性値が更新される。

表 6.1 は、チケットの一例を示したものである。タスクの進捗には時間の経過が伴うため、チケットデータは時間とともに属性値が変化するカテゴリデータに帰着させることができる。

表 6.1: チケットの一例

_At	Status	Priority	Assigned to	Product
2010/5/14	新規	通常	-	Application A
2010/5/15	新規	通常	Taro	Application A
2010/5/18	フィードバック	通常	Taro	Application A
2010/5/19	解決	通常	Taro	Application A
2010/5/21	終了	通常	Taro	Application A

6.1.1 分析に用いたデータ

研究室で使用しているプロジェクト管理ツールの redmine から、555 件のチケットデータを取得し、分析に用いた。データの期間は、2010 年 5 月 14 日から 2011 年 11 月 10 日までの 546 日間で、属性の数は 17 種類である。また、各属性値に対応するデータ件数の集計と River の描画は、一日を単位として行った。

¹<http://www.redmine.org/>

6.2 分析の実例

以下に、本ツールを用いた分析の一例を示す。

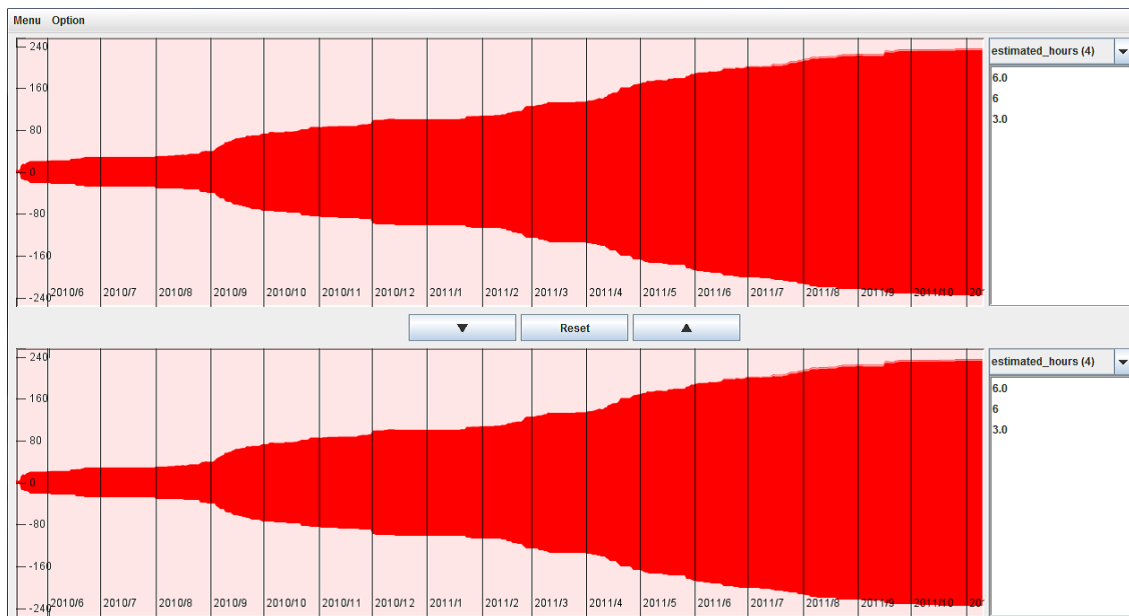


図 6.1: データを読み込んでから最初の表示

図 6.1 は、データを読み込んでから一番最初に表示される画面である。上下に描画された River を見ると、時間の経過によるデータ件数の変化が読み取れる。描画領域の右側ほど River の縦幅が広がっていることから、時間が進むにつれてデータが増加していることが分かる。

また、コンボボックスに表示されているテキストから、この River は `estimated_hours` 属性に対応していること、`estimated_hours` 属性は 4 種類の属性値を持っていることが分かる。コンボボックスの下にあるリストには、`estimated_hours` 属性が持つ属性値が表示されているが、文字で示されているのは 3 種類である。これは、`estimated_hours` 属性の値が指定されていないデータが存在するためである。このような場合でも、River には属性値なしの Stream が生成されている。

さらに、この River をマウスオーバーすると、第 5.2.5 節で述べたポップアップ機能により、各 Stream に対応する属性値が分かる。この例では、図 6.2 のようになり、River の大半が属性値なしの Stream で占められていることが分かる。Stream の縦幅は対応する属性値を持つデータの件数によるため、このデータセットにおけるデータのほとんどは、`estimated_hours` 属性を用いていないということを表している。したがって、この属性に着目した分析から、これ以上の有意な知見を得ることは難しいと思われる。

別の属性に着目した分析を行うために、コンボボックスから属性を選択して、表示する River を切り替える。ここでは、上側で `status` 属性を選択し、下側で `project` 属性を選択する。

図 6.3 は、切り替え操作を行った後の画面である。上下で異なる属性に対応する River を表

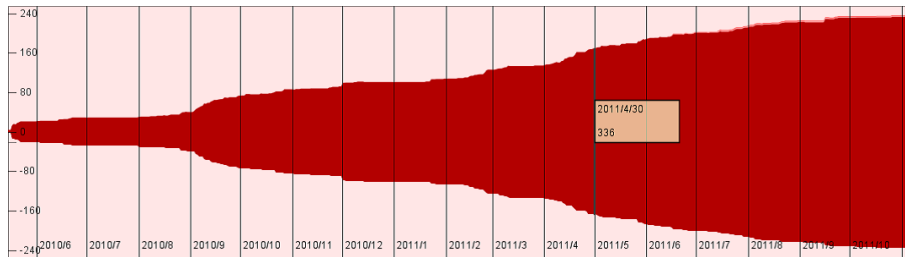


図 6.2: estimated_hours 属性の大半は、属性値なしであることが分かる

示しているため、River の色、Stream の形、表示される属性値がそれぞれ異なっている。属性値のリストを見ると、status 属性が持つ 6 種類の属性値をと、project 属性が持つ 7 種類の属性値が分かる。表示する River の属性を変えても同様に、各 Stream をマウスオーバー、または選択することで、Stream に何の属性値が対応しているかが分かる。

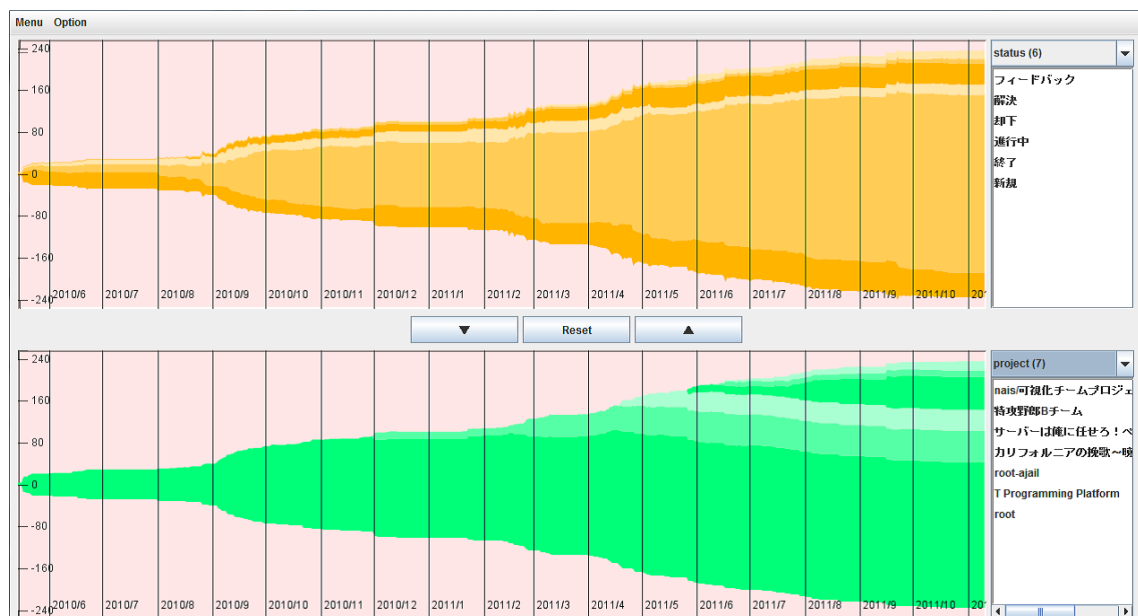


図 6.3: 上: status 属性、下: project 属性

図 6.3 の上側の River では、最初の方は「新規」の Stream が占める割合が多いが、時間が進むにつれて、「終了」の Stream が広くなる様子が分かる。また、「新規」、「進行中」、「フィードバック」、「解決」のデータは増加と減少を繰り返しているのに対して、「終了」、「却下」のデータは増加する一方であることが分かる。これは、チケットデータの status 属性に、「新規」→「進行中」、「フィードバック」、「解決」→「終了」、「却下」という値の変化の流れがあるからである。

下側の River では、時間が進むにつれて River の中に新しい Stream が発生し、上に積み上

がるような形になっていく様子が見て取れる。Stream の発生した位置は、その属性値に対応するデータが初めて作られた時を表している。このことから、それぞれに対応するプロジェクトが立ち上げられた時期が分かる。

また、それぞれの Stream の幅を見ることで、各プロジェクトに対応するデータの件数が比較できる。データの件数はタスクの量を表すため、それぞれのプロジェクトにかかっている労力が分かると言える。

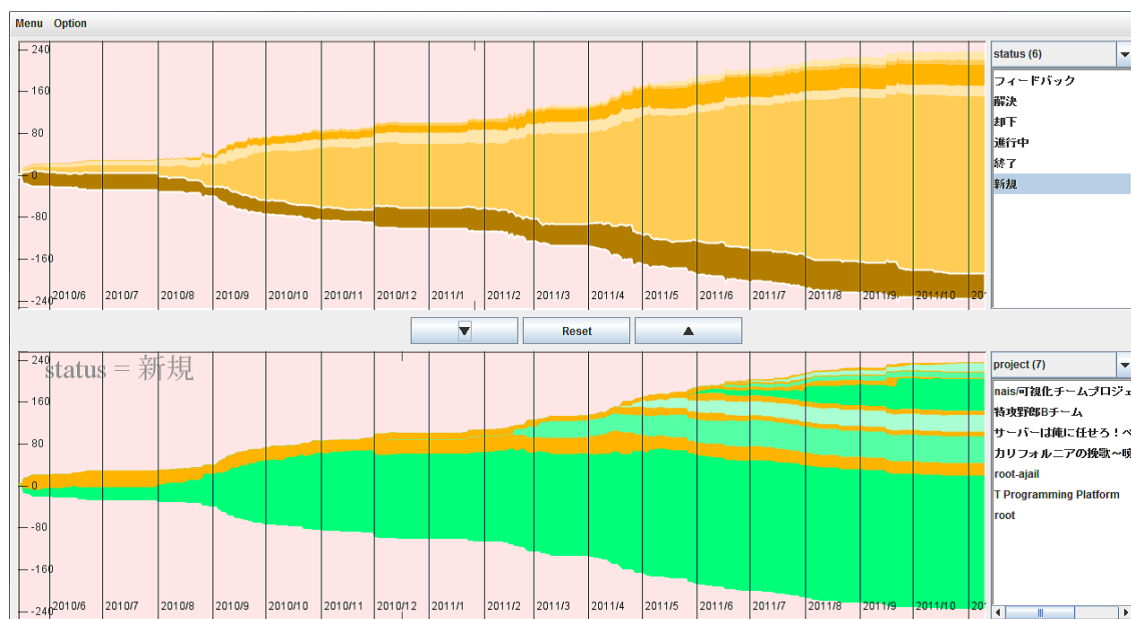


図 6.4: status 属性中の値「新規」の Stream を project 属性の River にオーバーレイ

さらに二つの属性の組み合わせに着目した分析を行うために、River オーバーレイを使用する。上側の River で「新規」の Stream を選択し、River オーバーレイを行うと、図 6.4 のようになる。これにより、「新規」の Stream が project 属性でどのように分布しているかが分かり、新たな知見が得られる。例えば、2010 年 12 月の頭に「新規」のデータが増加しているのは、新たなプロジェクトである「T Programming Platform」の立ち上げによるものだということが分かる。また、2011 年 9 月の半ばに「新規」のデータが減少しているのは、プロジェクト「カリフォルニアの挽歌」のデータが一気に更新されたためだということも読みとれる。

図 6.4 の状態から、下側の River の属性を切り替えて、assigned_to 属性を表示したものが図 6.5 である。第 5.2.4 節で述べたように、River オーバーレイ状態は表示する属性を切り替えても継続される。この状態では、「新規」の Stream の assigned_to 属性における分布を見ることができる。

図 6.5 からは、2011 年 4 月から 5 月を境にして、新規のデータの分布に変化があることが読み取れる。2011 年 4 月以前では、新規のデータは assigned_to 属性の値を持っていないものが多い傾向にある。しかし、そのようなデータは 2011 年 4 月以降、減少傾向にあることが分

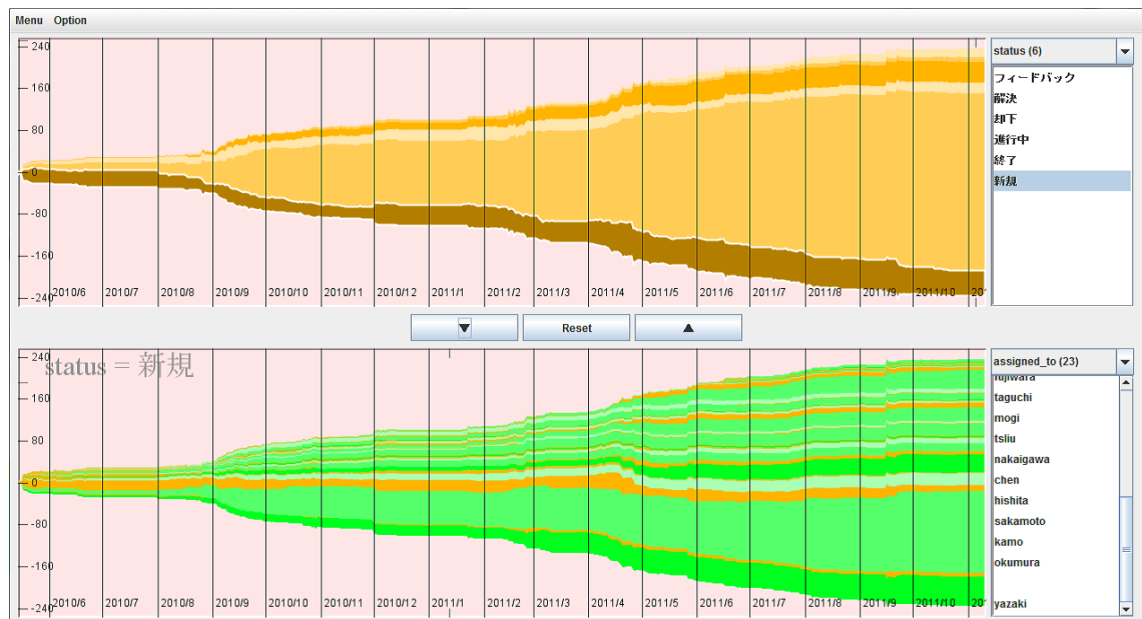


図 6.5: status 属性中の値「新規」の Stream を assigned_to 属性の River にオーバーレイ

かる。このことから、2011 年 4 月頃、チーム内で体制変更等の大きな動きがあったことが考えられる。

第7章 今後の課題

7.1 三属性以上の組み合わせの表現

本手法は、カテゴリデータの時間変化を視覚的に表現するにあたり、River オーバーレイによって、異なる二つの属性の組み合わせを表現することを可能としている。今後の拡張の一つとして、この対象を三つ以上の属性に広げることが考えられる。方法としては、一つの Stream に対して River オーバーレイを二つ以上表示し、属性値の対応に合わせてオーバーレイする Stream 同士を重ね合わせる方法などが考えられる。

7.2 大規模データへの対応

本研究で開発した手法は、数種類から数十種類までの属性や属性値を持つデータを対象としており、数千種類以上におよぶ属性や属性値を持つデータは対象としていない。このような大規模なデータへの対応を考えた際には、解決すべき新たな問題が発生すると思われる。考えられる問題としては、多くの属性値を表示することによる River や Stream の可読性の低下や、データ量の増大による計算コストの増加が挙げられる。可読性の問題については、隣接する複数の Stream を圧縮して表現する、一部の Stream を一時的に除外して描画する、などの解決策が考えられる。計算コストの問題については、高速に読み書きできるデータフォーマットの導入や、画面への描画を高速に行う拡張 API の導入によって、解決できると考える。

第8章 まとめ

本研究では、カテゴリデータの時間変化を視覚的に表現するための表現手法、および視覚的な分析ツールを開発した。本手法は、カテゴリデータの各属性と属性値を River と Stream によって表現し、さらに指定した属性に対応する River オーバーレイを用いることで、時間変化と異なる属性の組み合わせを同時に表現することが可能である。

また、本手法を用いた視覚的な分析ツール「TwinRiverViewer」を開発した。時間変化するカテゴリデータの分析要件を満たすために、二つの River を並べて表示するインターフェースや、River オーバーレイを行う為の機能を実装した。実データを読み込んで実際に分析を行うユースケースを提示し、本ツールを用いることによって知見が得られることを示した。

本研究では、カテゴリデータの時間変化を視覚的に表現することで、時間とともに変化するカテゴリデータから知見を得るための支援を行った。ユースケースにおいて扱ったチェックデータをはじめとして、時間変化するカテゴリデータとして表されるような情報は実世界に多く存在する。これら様々な情報に本手法を適用することにより、実世界の情報から新たな知見が得られる可能性がある。

謝辞

本研究を行うにあたり、指導教員である三末和男准教授をはじめ、田中二郎教授、志築文太郎講師、高橋伸准教授には、丁寧な指導と助言を頂きました。心から感謝を申し上げます。また、インタラクティブプログラミング研究室の皆様には、研究についての議論にとどまらず、研究室生活全体についての相談を聞いていただくなど、公私ともに大変お世話になりました。特に NAIS チームの皆様には、ゼミでの発表や議論などを通じてたくさんの貴重な意見を頂きました。迷惑をおかけしてしまうことも多々ありましたが、充実した研究室生活を送ることができたのは皆様のおかげです。深く感謝しております。最後に、家族や友人をはじめ、大学生活を送るにあたってお世話になった全ての方々に、心から感謝を申し上げます。本当にありがとうございました。

参考文献

- [1] Ben Fry 著, 増井 俊之 監訳, 加藤 慶彦 訳, “ビジュアライジング・データ—Processing による情報視覚化手法”, オライリー・ジャパン, 2008.
- [2] Susan Havre, Elizabeth Hetzler, Paul Whitney, and Lucy Nowell, “ThemeRiver: Visualizing Thematic Changes in Large Document Collections”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 8, No. 1, pp. 9–20, 2002.
- [3] J. Hartigan and B. Kleiner, “Mosaics for contingency tables”, *In Proceedings of the 13th Symposium on the Interface*, pp. 268–273, 1981.
- [4] Greg Smith, Mary Czerwinski, Brian Meyers, Daniel Robbins, George Robertson, and Desney S. Tan, “FacetMap: A Scalable Search and Browse Visualization”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 12, No. 5, pp. 797–804, 2006.
- [5] 白石 宏亮, 三末 和男, 田中 二郎, “つぶつぶ表現を用いたカテゴリデータの視覚的分析ツール”, インタラクション 2009 論文集, pp. 105–112, 2009.
- [6] Alfred Inselberg and Bernard Dimsdale, “PARALLEL COORDINATES: A TOOL FOR VISUALIZING MULTI-DIMENSIONAL GEOMETRY”, *Proceedings of the First IEEE Conference on Visualization '90*, pp. 361–378, 1990.
- [7] Robert Kosara, Fabian Bendix, and Helwig Hauser, “Parallel Sets: Interactive Exploration and Visual Analysis of Categorical Data”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 12, No. 4, pp. 1–12, 2006.
- [8] Christopher Collins, Fernanda B. Viegas, and Martin Wattenberg, “Parallel Tag Clouds to Explore and Analyze Faceted Text Corpora”, *IEEE Symposium on Visual Analytics Science and Technology*, pp. 91–98, 2009.
- [9] Milos Krstajic, Enrico Bertini, and Daniel Keim, “CloudLines: Compact Display of Event Episodes in Multiple Time-Series”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 12, pp. 2432–2439, 2011.
- [10] Marc Weber, Marc Alexa, and Wolfgang Muller, “Visualizing Time-Series on Spirals”, *Proceedings of the IEEE Symposium on Information Visualization 2001*, pp. 7–13, 2001.

- [11] Marian Dork, Daniel Gruen, Carey Williamson, and Sheelagh Carpendale, “A Visual Backchannel for Large-Scale Events”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 16, No. 6, pp. 1129–1138, 2010.
- [12] Weiwei Cui, Shixia Liu, Li Tan, Conglei Shi, Yangqiu Song, Zekai J. Gao, Xin Tong, and Huamin Qu, “TextFlow: Towards Better Understanding of Evolving Topics in Text”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 12, pp. 2412–2421, 2011.
- [13] Zhicheng Liu, John Stasko, and Timothy Sullivan, “SellTrend: Inter-Attribute Visual Analysis of Temporal Transaction Data”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 15, No. 6, pp. 1025–1032, 2009.
- [14] 矢崎 聖也, 三末 和男, 田中 二郎, “ソフトウェア開発プロジェクトの探索的分析を支援する視覚的分析ツール”, 日本ソフトウェア科学会第 27 回大会, 2010.